**Quiz 1**

**Muhammad Haroon Shahzad**

**SP21-BCS-017**

Introduction to Data Science SP2024

Marks 5

Due date: 20-March-2024 at 11:30 AM

Question: Create animated graphs by using matplotlib library. At least 3 charts are required. Paste the code, date and screen shot of the charts. **Important**: Fill the data and print the output. Paste the code and screenshots in submission. Strictly not more than one students per submission.

**Declaration:** I/We declare that:

1. I have created at least 50% of the answers by my own (not copied and without any help from any source).
2. I have executed the code at least once.
3. There are no bug/s in the code.
4. I understand the working of the code.

Signature/s: …………………………………

Roll No./s: sp21-bcs-017

**Answer**: (Code, data and screen shot/s)

Import necessary Libraries.

import matplotlib.pyplot as plt

import matplotlib.animation as animation

import numpy as np

create sample data for animation

x = np.linspace(0, 10, 100)

y1 = np.sin(x)

y2 = np.cos(x)

y3 = np.random.rand(100) \* 10

Functions of different Chart

def animate\_line(i):

    plt.clf()

    plt.plot(x[:i], y1[:i], color='blue', marker='o')

    plt.title('Line Chart')

    plt.xlabel('X-axis')

    plt.ylabel('Y-axis')

    plt.text(7, 0, f'Data: x={x[i]:.2f}, y={y1[i]:.2f}', fontsize=10)

def animate\_bar(i):

    plt.clf()

    plt.bar(x[:i], y2[:i], color='green')

    plt.title('Bar Chart')

    plt.xlabel('X-axis')

    plt.ylabel('Y-axis')

    plt.text(7, 0, f'Data: x={x[i]:.2f}, y={y2[i]:.2f}', fontsize=10)

def animate\_scatter(i):

    plt.clf()

    plt.scatter(x[:i], y3[:i], color='red', marker='s')

    plt.title('Scatter Plot')

    plt.xlabel('X-axis')

    plt.ylabel('Y-axis')

    plt.text(7, 2, f'Data: x={x[i]:.2f}, y={y3[i]:.2f}', fontsize=10)

Create Animation and save as GIF

ani\_line = animation.FuncAnimation(plt.gcf(), animate\_line, frames=len(x), interval=50)

ani\_bar = animation.FuncAnimation(plt.gcf(), animate\_bar, frames=len(x), interval=50)

ani\_scatter = animation.FuncAnimation(plt.gcf(), animate\_scatter, frames=len(x), interval=50)

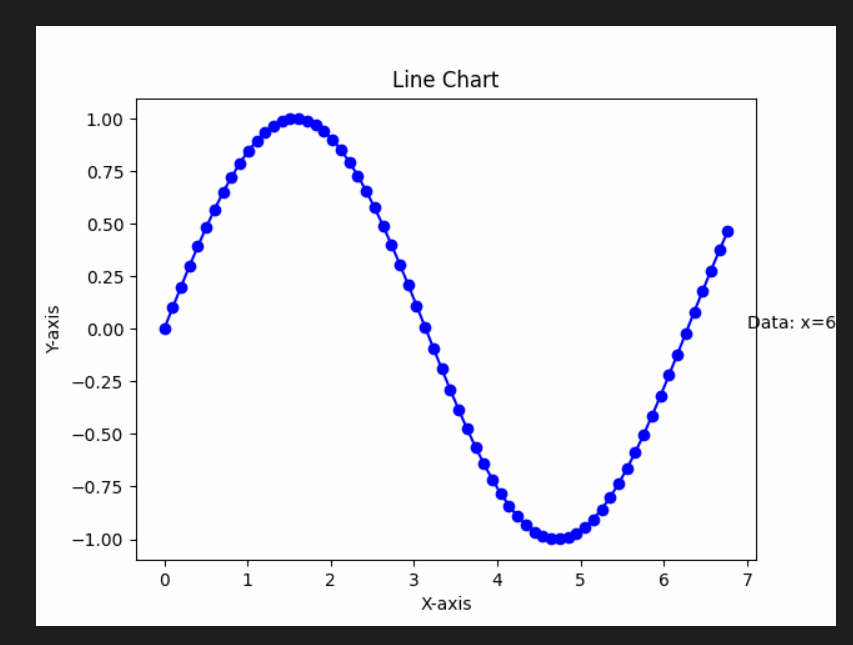
ani\_line.save('line\_chart\_animation.gif', writer='pillow')

ani\_bar.save('bar\_chart\_animation.gif', writer='pillow')

ani\_scatter.save('scatter\_plot\_animation.gif', writer='pillow')

plt.show()

PLOTS



A green graph with black text

Description automatically generated

A screen shot of a graph

Description automatically generated

Gifs of plots

A graph with lines and numbers

Description automatically generated with medium confidence

A white rectangular graph with black text

Description automatically generated

A graph with numbers and lines

Description automatically generated with medium confidence